



MANUAL USUARIO

**Universidad Mariano Gálvez de Guatemala
INGENIERÍA EN SISTEMAS DE INFORMACIÓN Y
CIENCIAS DE LA COMPUTACIÓN
Programación III**

Integrantes del grupo:

Integrantes:	Carnet:
Josué Vinicio Jerez Gómez	9490-22-1479
Mario Roberto Rompich Yoc	9490-17-17052

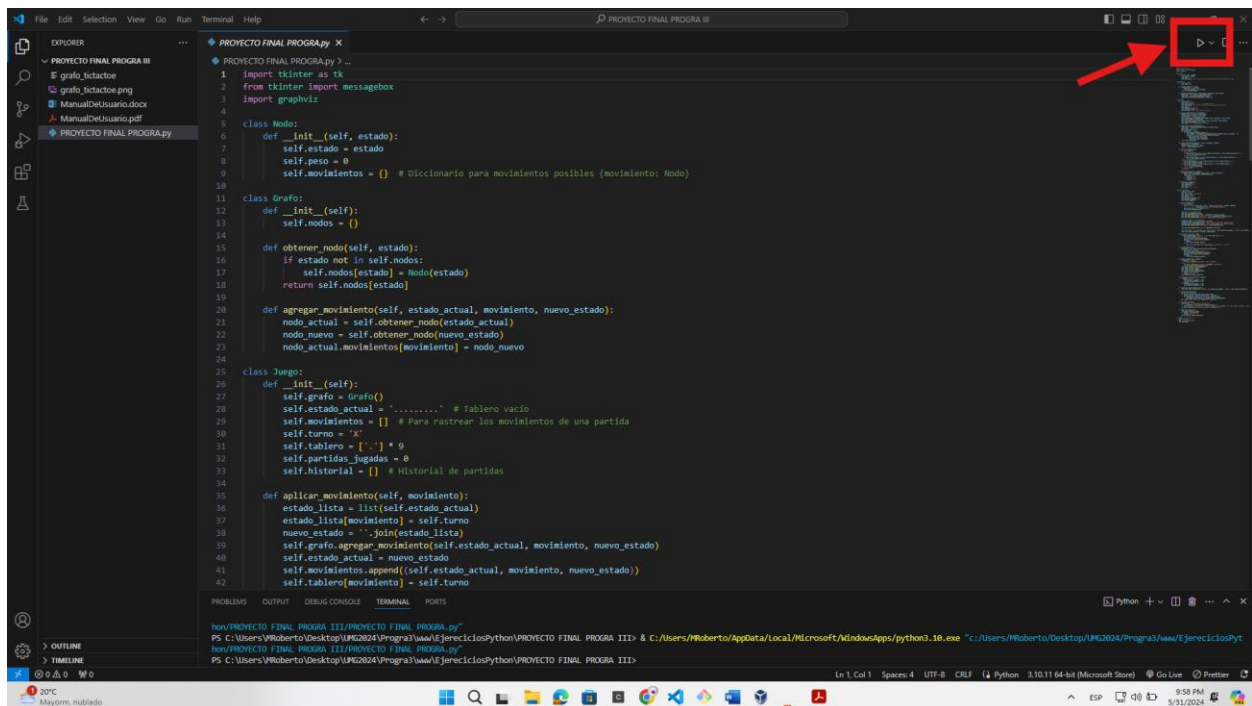
A continuación, se mostrarán capturas e instrucciones que le ayudarán al usuario a levantar y utilizar el proyecto de totito inteligente. El código fuente se encuentra en el siguiente repositorio el cual tendremos que descargar o clonar el repositorio.

<https://github.com/JosueJerez/TAREAS-Y-PROYECTOS-PROGRA-III/blob/main/PROYECTO%20FINAL%20PROGRA%20III/PROYECTO%20FINAL%20PROGRA.py>

El programa fue desarrollado en el lenguaje de programación Python para correr el programa necesitaremos descargar e instalar Python en nuestra computadora en este caso la instalación se hará en una computadora con sistema operativo windows. En el siguiente enlace podremos descargar Python <https://www.python.org/downloads/windows/> y luego procedemos a instalarlos en nuestra computadora

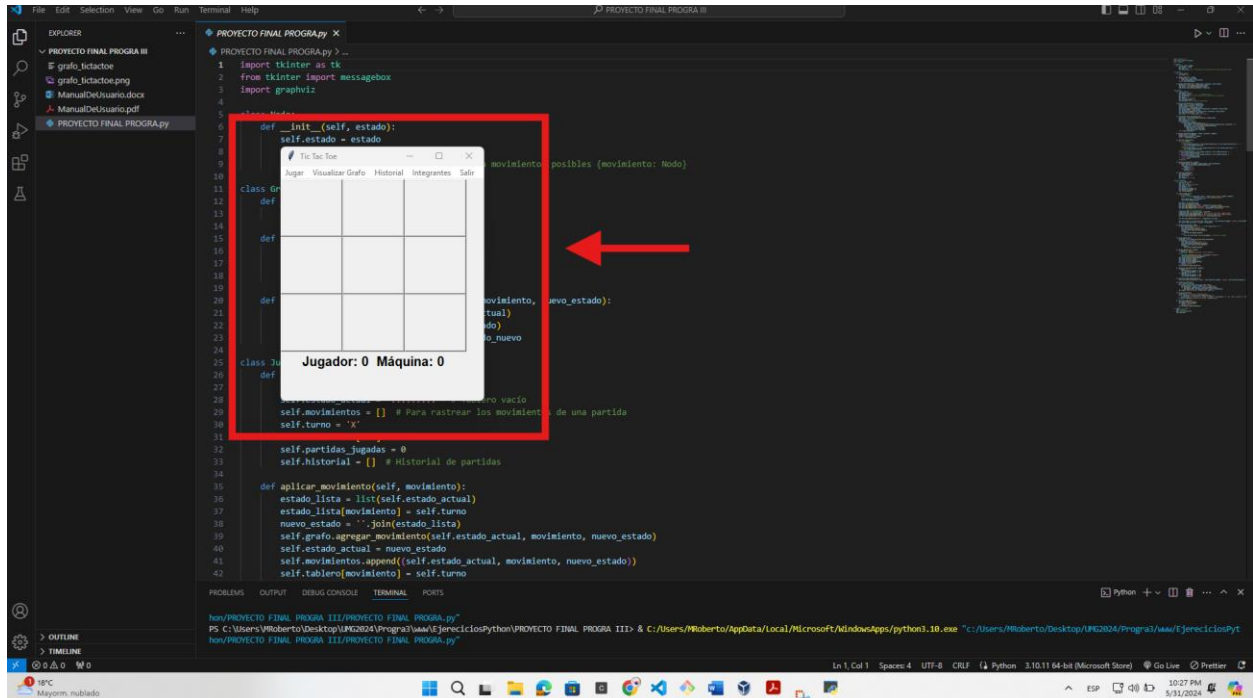
Luego procedemos a instalar una librería externa que sería Graphviz que se puede descargar en el siguiente enlace <https://graphviz.org/download/> para representar la gráfica de la estructura de datos utilizada.

Una vez instaladas las herramientas procedemos a correr el código.

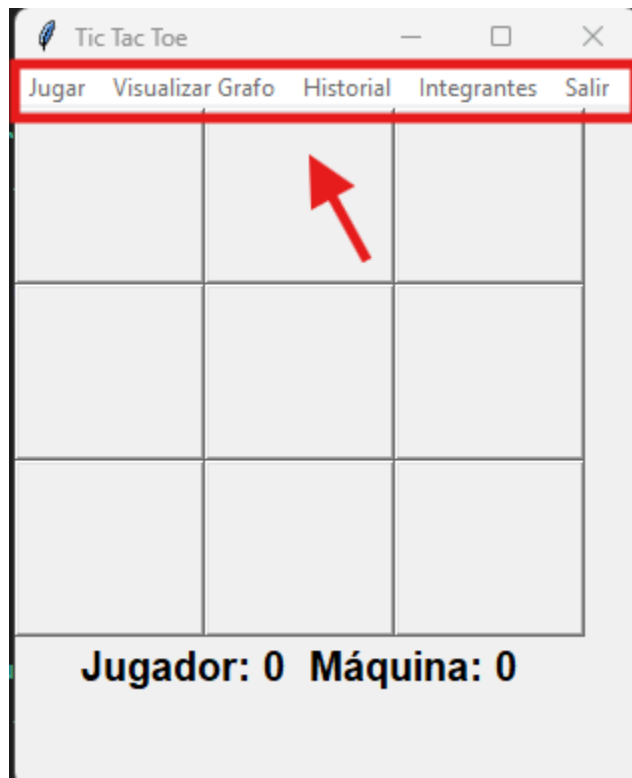


```
1 import tkinter as tk
2 from tkinter import messagebox
3 import graphviz
4
5 class Nodo:
6     def __init__(self, estado):
7         self.estado = estado
8         self.peso = 0
9         self.movimientos = {} # Diccionario para movimientos posibles (movimiento: Nodo)
10
11 class Grafo:
12     def __init__(self):
13         self.nodos = {}
14
15     def obtener_nodo(self, estado):
16         if estado not in self.nodos:
17             self.nodos[estado] = Nodo(estado)
18         return self.nodos[estado]
19
20     def agregar_movimiento(self, estado_actual, movimiento, nuevo_estado):
21         nodo_actual = self.obtener_nodo(estado_actual)
22         nodo_nuevo = self.obtener_nodo(nuevo_estado)
23         nodo_actual.movimientos[movimiento] = nodo_nuevo
24
25 class Juego:
26     def __init__(self):
27         self.grafo = Grafo()
28         self.estado_actual = '.....' # Tablero vacío
29         self.movimientos = [] # Para rastrear los movimientos de una partida
30         self.turno = 'X'
31         self.tablero = ['.'] * 9
32         self.partidas_jugadas = 0
33         self.historial = [] # Historial de partidas
34
35     def aplicar_movimiento(self, movimiento):
36         estado_lista = list(self.estado_actual)
37         estado_lista[movimiento] = self.turno
38         nuevo_estado = ''.join(estado_lista)
39         self.grafo.agregar_movimiento(self.estado_actual, movimiento, nuevo_estado)
40         self.estado_actual = nuevo_estado
41         self.movimientos.append((self.estado_actual, movimiento, nuevo_estado))
42         self.tablero[movimiento] = self.turno
```

Entonces se nos abrirá la siguiente ventana:



En la ventana abierta se nos muestra un menú el cual podemos interactuar con el.



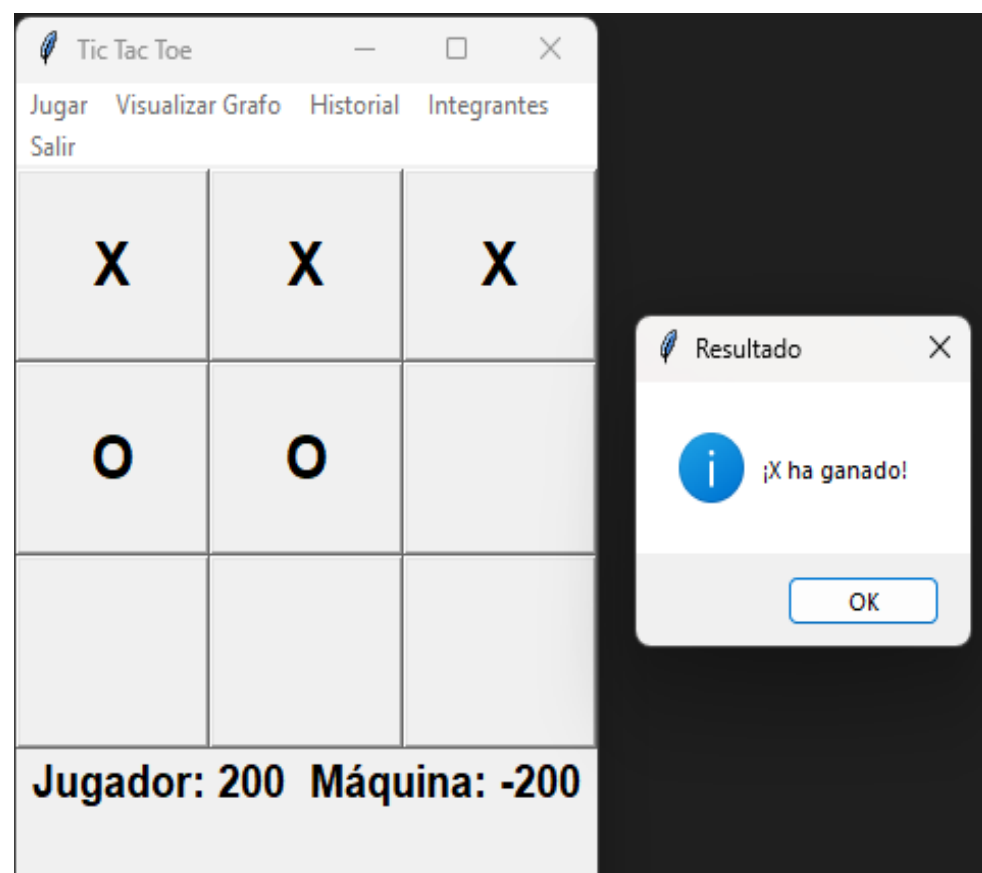
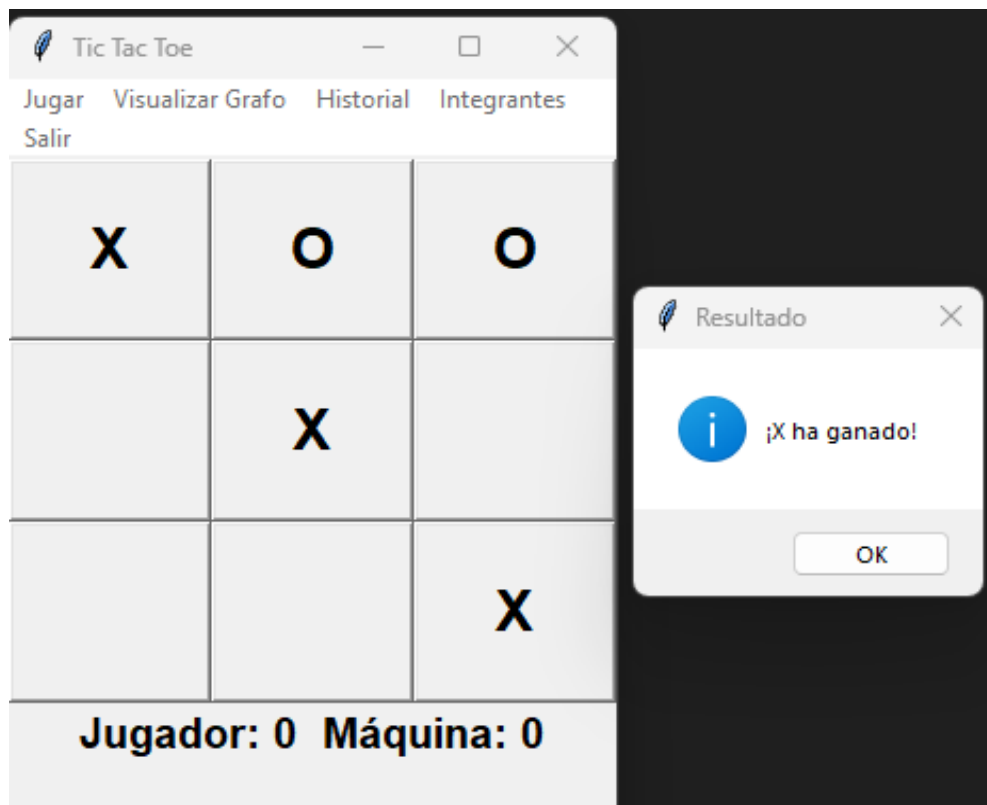
Cuando elegimos la opción jugar, iniciamos el juego en donde el usuario iniciar eligiendo la primera casilla y estaría marcando con una (X) y la maquina automáticamente respondería con la mejor posición con un (O).

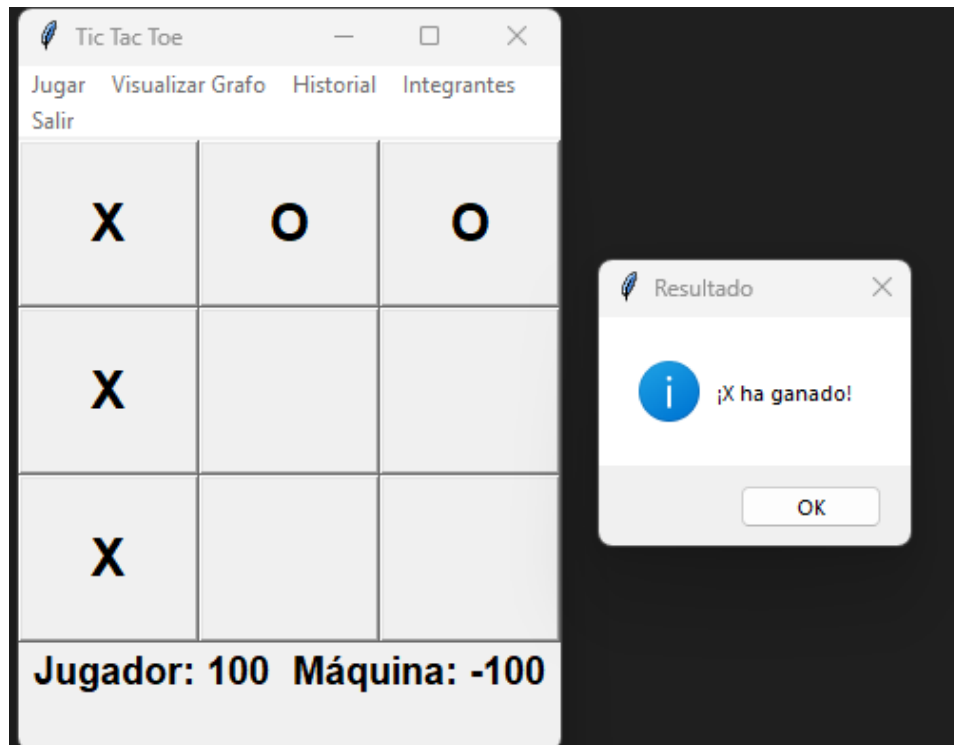


Cuando el usuario ó máquina tenga las tres casillas llenas con (X) ó (O) ya sea en línea diagonal, horizontal o vertical, el primero que complete las tres casillas será el ganador de la partida.

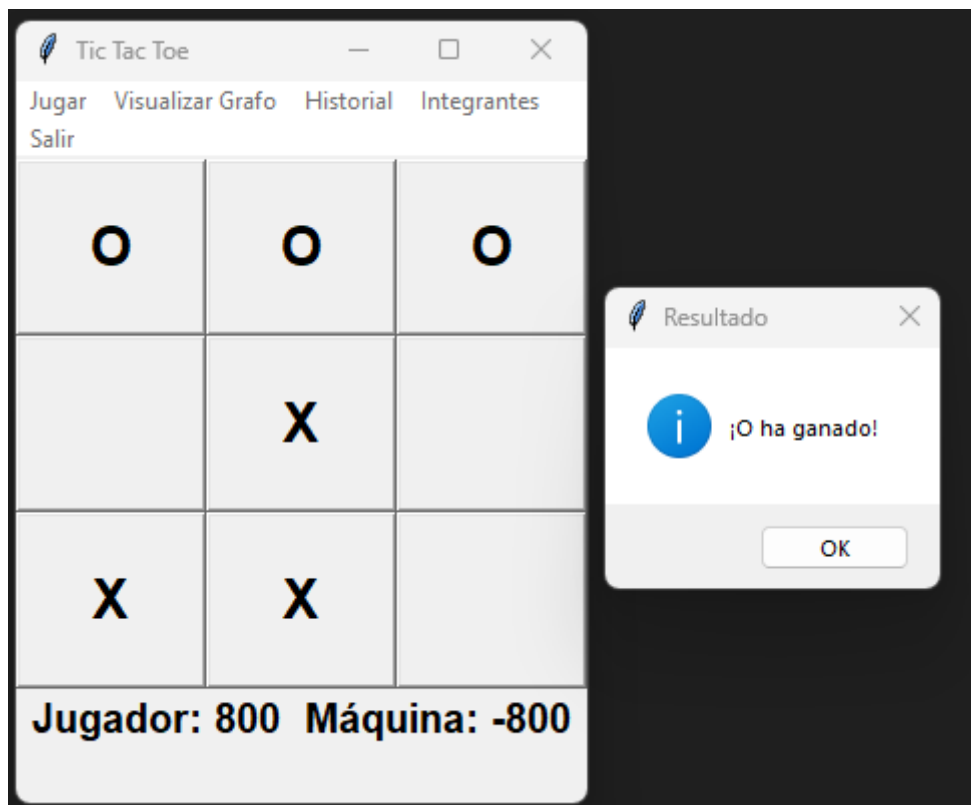
Como podemos observar en las siguientes capturas el usuario ha ganado las tres partidas. La primera partida la gano en la línea vertical, la segunda partida en línea horizontal y la tercera partida en línea vertical. En donde se le suman o restan 100pts al usuario o la máquina que dependiendo quien gane o pierda la partida.

Sin embargo, la maquina estaría aprendiendo entre mas partidas se jueguen. Para que las siguientes partidas elija la mejor casilla y así poder ganarle al usuario.

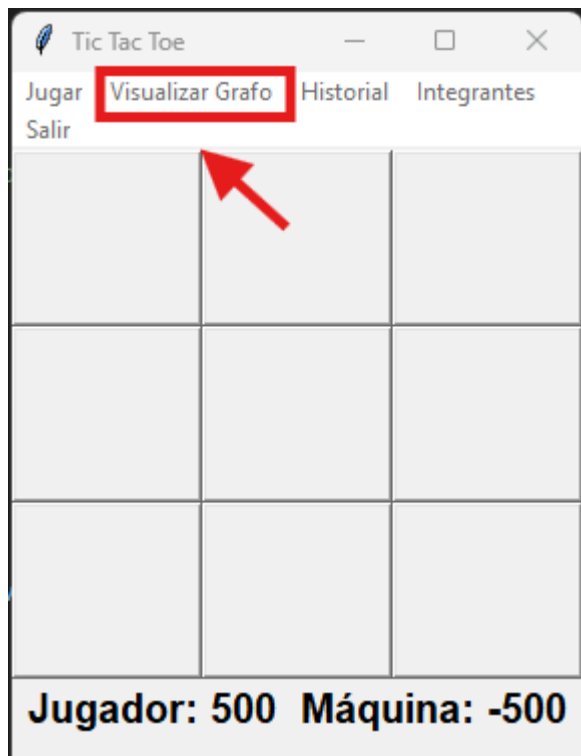




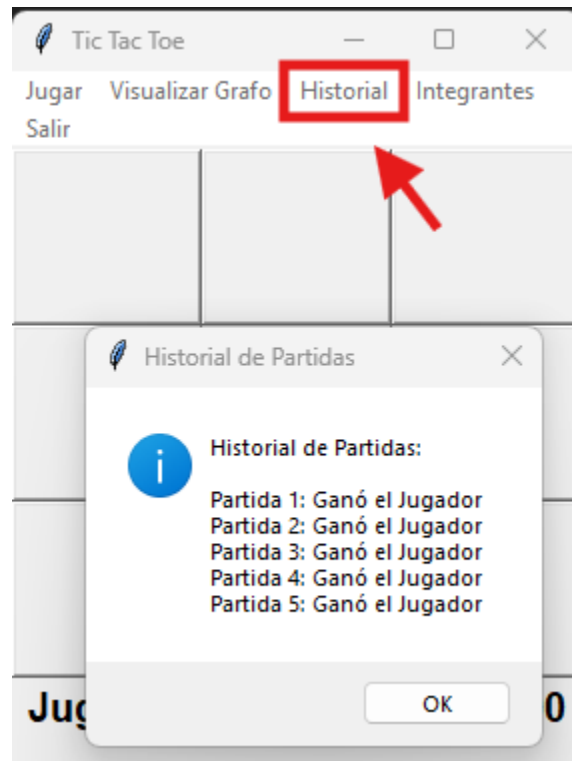
Como podemos observar después de unas cuantas partidas la maquina ha aprendido y nos ha ganado la partida.



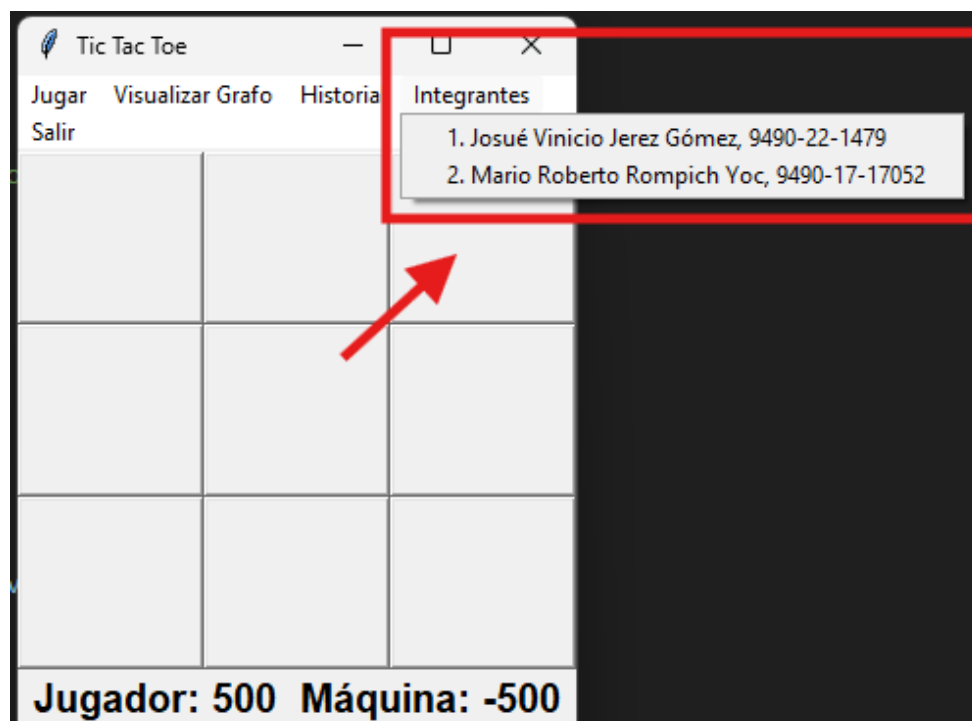
Siguiendo con las opciones del menú encontramos la opción de visualizar grafo. Esta opción nos despliega una imagen en formato png con el grafo que se va construyendo con respecto las partidas.



En la sección de Historial podremos observar el historial de partidas que el usuario y la maquina han ganado.



En la siguiente opción nos muestra la informa de integrantes de las personas que participaron en el desarrollo del proyecto.



En la última opción estaría la opción de salir del programa o cerrarlo.

